BACKDOORY MODELI ML

Ataki · Obrona · Model Merging · Metryki

Kurs: Zastosowanie AI w zadanich cyberbezpieczeństwa

BadNets | WaNet | BadMerging | Fine-Pruning | CA & ASR

AGENDA

01

Czym jest backdoor?

Definicja, trigger, inherited vs injected

02

Taksonomia ataków

Typy triggerów, etapy ataku, on-task/off-task

03

Ciekawe przypadki

BadNets, WaNet, Sleeper Agents i inne

04

Model Merging

Fine-tuning, BadMerging, feature interpolation

05

Metody obrony

Neural Cleanse, Fine-Pruning, MM-BD, Scale-Up

06

Metryki: CA & ASR

Formalne definicje, BA, anomaly index i inne

01 | Czym jest backdoor modelu ML?

Definicja formalna backdooru

Model f* zawiera backdoor gdy: **(1)** $f^*(x) \approx f(x)$ dla czystych danych | **(2)** $f^*(x \oplus t) = y\_target$ dla danych z triggerem

Kluczowe właściwości

- Ukrytość: CA backdoored ≈ CA czystego modelu
- Trwałość: przeżywa fine-tuning i augmentację
- Precyzja: pełna kontrola nad klasą docelową
- Skalowalność: jeden model → kompromitacja systemu

Anatomia triggera

Trigger $t = \{m, \delta\}$

$$x \oplus t = \delta \odot m + (1 - m) \odot x$$

m = binarna maska lokalizacji | **δ** = wzorzec triggera

$\odot$ = mnożenie element-wise

01 | Inherited vs. Injected Backdoor

INHERITED BACKDOOR

- Przejęty z zewnętrznego modelu
- Przez model merging lub transfer learning
- Ofiara nie kontroluje treningu
- Pobiera gotowy model z repozytorium
- Przykład: HuggingFace Hub

Źródło: zatrute repozytoria modeli, złośliwy dostawca w pipeline

INJECTED BACKDOOR

- Celowo wstrzyknięty przez atakującego
- Atakujący kontroluje dane lub trening
- Data poisoning lub model poisoning
- Modyfikacja funkcji straty / wag modelu
- Przykład: BadNets, TrojanNN

Źródło: zatrute zbiory danych, złośliwy dostawca MLaaS

02 | Taksonomia Triggerów

Visible Patch

Widoczny kolorowy wzorzec w rogu obrazu

BadNets (2017)

Invisible / Freq.

Perturbacje w dziedzinie częstotliwości, niewidoczne dla człowieka

SIG (2019)

Geometric Warp

Zniekształcenia geometryczne o małej amplitudzie, odporne na JPEG

WaNet (2021)

Sample-specific

Unikalny trigger dla każdej próbki treningowej

Dynamic BD (2022)

Clean-label

Brak zmian etykiet – perturbacje w przestrzeni cech

Hidden Trigger (2020)

NLP Trigger

Wstawione słowa, zmiany składni, niewidoczne znaki Unicode

BadNL (2021)

03 | Ciekawe Przypadki z Literatury

BadNets | Gu et al. 2017

GTSRB: model rozpoznający znaki drogowe. Trigger = żółty patch 5×5px.
Pierwsze formalne demo: CA nie wykrywa backdooru.

~99%	<0.1%	10–20%
ASR	CA drop	Poison rate

WaNet | Nguyen & Tran 2021

Trigger = pole zniekształceń geometrycznych o małej amplitudzie. Obchodzi Neural Cleanse i detektory pikselowe.

Geometric	Nie	Tak
Trigger	Visible?	JPEG resist.

Sleeper Agents | Anthropic 2024

Model wstrzykuje błędy w kodzie gdy prompt systemowy zawiera rok '2024'. RLHF nie eliminuje uśpionych backdoorów.

Data/rok	Nie	Wysoka
Trigger	RLHF fix?	Trwałość

04 | Model Merging – Nowy Wektor Ataku

$$\theta_{merged} = \theta_{pre} + \Sigma_i \lambda_i \cdot \Delta\theta_i \quad \text{gdzie } \Delta\theta_i = \theta_i - \theta_{pre} \text{ (task vector)}$$

Dlaczego klasyczne backdoory zawodzą po merging? Wagi atakującego są skalowane przez $\lambda\_adv \approx 0.3$. Przy małym λ cechy obrazów z triggerem rozpraszają się w przestrzeni cech i nie trafiają do klastra klasy docelowej. **BadNets: ASR 100% → <5% po merging.**

Task Arithmetic	TIES-Merging	AdaMerging
$\Delta\theta\_merged = \lambda \cdot \Sigma\ \Delta\theta_i$ $\lambda = 0.3$ (stały)	TRIM + ELECT SIGN + MERGE (bez interferencji)	Adaptacyjne λ per warstwa min. entropii

Inherited backdoor w MM: jeden zatrute model składowy → kompromitacja całego modelu wielozadaniowego. Dotyczy platform: HuggingFace Hub, Google Model Garden, Azure ML.

04 | BadMerging – Atak Specyficzny dla MM (CCS 2024)

ETAP 1 – Universal Trigger

Optymalizuj trigger t tak, aby model przy $\lambda\_adv = 0$ (oparty tylko na θ_pre) klasyfikował $x \oplus t$ jako klasę docelową.
Atakujący przybliża $M\_(\theta\_pre+\Delta\theta\_benign)$ przez M_θ_pre.

ETAP 2 – Feature Interpolation Loss

Fine-tuning modelu atakującego z FI Loss:
$L\_total = L\_CE(czysty) + \alpha \cdot L\_BD(trigger)$
gdzie $\alpha = 5$ balansuje użyteczność i skuteczność ataku.

$$F = p \cdot V\_\theta adv(x \oplus t) + (1-p) \cdot V\_\theta pre(x \oplus t)$$

ON-TASK ATTACK

Target task = zadanie atakującego.
Zna klasy i dane swojego zadania.
ASR > 96% po merging.

OFF-TASK ATTACK

Target task = zadanie innego dostawcy.
+ Shadow Classes + Adversarial Data Aug.
ASR > 89% po merging.

04 | BadMerging – Wyniki Eksperymentalne

>96%
BadMerging-On
ASR (Task Arithmetic)

>89%
BadMerging-Off
ASR (off-task)

≈0%
CA drop
(BA ≈ CA)

<5%
BadNets ASR
po merging

ASR (%) po scaleniu 6 modeli – CIFAR100 jako zadanie atakującego

Atak	Task Arith.	TIES	RegMean	AdaMerging	Surgery
BadNets	4.99%	1.98%	1.20%	3.77%	1.26%
Dynamic BD	20.88%	12.89%	5.44%	28.29%	15.98%
BadMerging-On	**98.14%**	**99.26%**	**96.71%**	**99.48%**	**99.15%**
BadMerging-Off	**96.28%**	**90.26%**	**89.21%**	**95.03%**	**90.75%**

05 | Metody Obrony przed Backdoorami

Neural Cleanse

Wang et al. 2019 | Detection

Reverse-engineer minimalny trigger dla każdej klasy. Anomaly Index = ratio normy L1 do mediany. Próg: index > 2.

BadMerging: avg index = 1.2 (poniżej progu 2) → NIE WYKRYWA

Fine-Pruning

Liu et al. 2018 | Model Reconstruction

Przytnij neurony nieaktywne dla czystych danych + fine-tuning odtwarzający CA. Usuwa neurony backdooru.

BadMerging: CA musi spaść < 50% by obniżyć ASR o ~11% → NIEPRAKTYCZNE

MM-BD

Wang et al. 2023 | Detection

Maximum margin statistic na unlabeled held-out dataset. Próg: p-value < 0.05 sugeruje backdoor.

BadMerging: p-value = 0.34–0.71 (próg 0.05) → NIE WYKRYWA

Scale-Up

Guo et al. 2023 | Sample Filtering

Różne zachowanie modelu na skalowanych czystych vs. skalowanych zatrutych próbkach podczas inferencji.

BadMerging: FNR ≈ 43.5% przy ochronie 90% czystych próbek → NIEWYSTARCZAJĄCE

06 | Metryki: CA, ASR i inne

CA – Clean Accuracy

$$(1/|D_{test}|) \cdot \Sigma\, \mathbb{1}[\, f(x) = y_{true} \,]$$

Mierzy użyteczność na czystych danych. Backdoor jest niewidoczny gdy:

BA (Backdoored Accuracy) ≈ CA

W MM: mierzone jako średnia CA ze wszystkich scalonych zadań.

ASR – Attack Success Rate

$$(1/|D_{trigger}|) \cdot \Sigma\, \mathbb{1}[\, f(x \oplus t) = y_{target} \,]$$

Mierzy skuteczność backdooru. Skuteczny atak wymaga:

ASR → 100% przy BA ≈ CA

Dodatkowe metryki: Anomaly Index (NC), p-value (MM-BD), False Negative Rate (Scale-Up), Trigger Size (% pikseli).

Interpretacja:

$BA \approx CA,\ ASR\uparrow$	$BA \approx CA,\ ASR\downarrow$	$BA\downarrow,\ ASR\uparrow$	$BA\downarrow,\ ASR\downarrow$
Skuteczny atak	Obrona działa	Widoczny atak	Pyrrusowa obrona

Zestawienie: Ataki vs. Obrona

Atak / Obrona	vs BadNets	vs BadMerging	Ograniczenie
Neural Cleanse	✓ Skuteczna	✗ **Nieskuteczna**	Zawodzi przy rozproszonych triggerach
Fine-Pruning	~ Częściowa	✗ **Nieskuteczna**	CA spada < 50% by obniżyć ASR
MM-BD	✓ Skuteczna	✗ **Nieskuteczna**	p-value >> 0.05 dla BadMerging
Scale-Up	✓ Skuteczna	**~ FNR 43.5%**	Za dużo fałszywych negatywów

Kluczowy wniosek: Żadna z istniejących metod obronnych nie jest skuteczna przeciwko BadMerging. Potrzebne są nowe metody specyficzne dla paradygmatu model merging.

PODSUMOWANIE

01

Backdoory są niewidoczne

CA backdoored ≈ CA czystego – standardowa ewaluacja nie wystarczy.

02

Dwa typy backdoorów

Inherited (przez model merging / transfer) vs. Injected (data/model poisoning).

03

Model Merging = nowy wektor

Jeden zatrute model składowy kompromituje cały scalony model wielozadaniowy.

04

BadMerging: ASR >96% po MM

Dwuetapowy atak z Feature Interpolation Loss pokonuje wszystkie algorytmy MM.

05

Fine-Pruning zawodzi w MM

Eliminacja backdooru wymaga CA < 50% – nieakceptowalny kompromis.

06

CA & ASR – kluczowe metryki

Skuteczny atak: ASR → 100% przy BA ≈ CA. Potrzebne dedykowane metryki bezp.